In *Proceedings of the 4th International Workshop on the Design, Specification, and Verification on Interactive Systems*, Granada, Spain, 1997.

Human-System Modeling:
Some Principles and a Pragmatic Approach

Michael A. Freed & Michael G. Shafto

Cognition Research Group
Human-Automation Integration Research Branch
NASA-Ames Research Center
mfreed@mail.arc.nasa.gov
mshafto@mail.arc.nasa.gov

Abstract

There are a number of formalisms and architectures for modeling human performance, but there is little guidance on how to go about building useful human models. This is a serious problem since human modeling is difficult and full of pitfalls. The intended application of the model should play a strong guiding role in model development. We are building an engineering model intended to help interface designers predict usability problems -- in particular, to alert designers to features of an interface that may increase the risk of certain kinds of human error. In reflecting on our experience in building this model, we have developed several inter-related principles that have been helpful in directing our investment of time and effort. Taken together, these principles suggest a methodology for the development of human performance models for complex human-machine systems.

Introduction

When tasks make excessive demands on cognitive and perceptual resources, people often make mistakes. In many cases, such mistakes result from a failure to take human capabilities into account when designing the procedures and equipment to be used in carrying out the task. Good design thus requires methods for determining whether procedures and equipment inadvertently facilitate error.

Unfortunately, existing methods for determining compatibility between a human operator and the procedures and equipment he or she will use are far from ideal. Analytical methods, such as those often found in handbooks, guideline documents, or proposed standards, use ad hoc mixtures of domain knowledge, psychological findings, common sense, and general human factors design guidelines to predict usability problems. Analytical methods are inexpensive, and they can be employed at an early stage in a design process. Therefore, recommendations made on the basis of such sources may be implementable at relatively low cost. However, analytical methods are largely ineffective at predicting the performance impact of new equipment and procedures for dynamic, complex tasks under highly variable operating conditions.

Another possibility is to test human subjects using the new procedures and equipment in a simulated task environment. Empirical user testing [Gould88] is effective, since actual performance with proposed equipment and procedures is directly observed and measured. However, this method tends to be expensive, since any proposed equipment must actually be prototyped in some detail and human subjects must be recruited. In air traffic control, the need to hire expert subjects for long periods of time limits the number of designs that can be examined and inhibits the number of refinements that can be tested. It is difficult to find enough expert subjects in domains like air traffic control to test adequately even one proposed design [Remington90a, Shafto90].

Some of the benefits of an empirical study may be obtained at lower cost by using a computer to simulate task logic, psychological resources, and environmental constraints. Human simulation has been used successfully by others to guide design. For example, John & Gray [Gray93, John94] simulated telephone operators to predict the time-savings that would be achieved with a new workstation design. The simulation accurately predicted that the new equipment would actually increase the amount of time required to handle calls, thus potentially saving the telephone company from a costly mistake.

Seifert and Shafto [Seifert94] surveyed a variety of approaches to cognitive modeling. They concluded that different modeling approaches focus to differing degrees on three major sources of constraints: task logic, psychological resources, and environmental structure. In air traffic control, task logic includes the design of equipment and procedures; psychological resources include task-specific knowledge as well as general cognitive, perceptual, and motor resources; and environmental structure includes both physical and social factors.

No single theory has succeeded in unifying all these types of factors. Perhaps the most ambitious practical attempt has been the MIDAS system [Corker93], which includes some capability in all three areas and is unusually strong in representing environmental structure. The AMODEUS Project [AMODEUS97] includes a number of methodologies [May94] with some attention to formal system-modeling methods for integrating different classes of constraints. Degani and his colleagues [Degani96, Degani97] have also demonstrated the use of formal methods to model procedural and environmental constraints, although their approach has not yet been extended to the modeling of psychological resources. The Cognition Simulation System [CSS, Remington90] represents an approach that is complementary to that of Degani et al., focusing on psychological resources and excluding detailed treatment of task logic and environmental structure.

Despite these promising lines of current research, many experienced researchers and practitioners [Landauer95, for example] remain skeptical about the contribution that human-system models can make to the solution of practical design problems. Such skepticism is appropriate in the absence of clear demonstrations to the contrary. In any case, to be useful in the analysis of complex real-world tasks such as air traffic control, human-system models must provide a way to integrate constraints on performance arising from the sources mentioned above. In specific cases, information about these classes of constraints may be differentially available. For example, detailed knowledge of perceptual, motor, attentional, and memory resources may be available, but little information may be initially available about the strategies typically used by experts to overcome these inherent limitations. Further, certain information related to the physical ergonomics of the task environment may be readily available, while the less tangible constraints of the social and communications environment remain largely unknown. For this reason, we need not only an architecture which can integrate multiple kinds of constraints, but also a methodology which proceeds according to a refinement strategy [Shafto94], first incorporating those constraints which are best understood and which most obviously contribute to variability in task performance or vulnerability to error.

Our System: Predicting Design-facilitated Error

While there are a number of formalisms and architectures for modeling human performance, there is very little guidance on how to go about building useful human models. This is a serious problem since human modeling is difficult and full of pitfalls.

The intention to apply a model to help analyze designs should strongly constrain how the model is constructed.

We are building an engineering model intended to alert interface designers to features of an interface that may increase the risk of certain kinds of human error. In reflecting on our experience in building this model, we have developed several inter-related principles that have been helpful in directing our investment of time and effort. Taken together, these principles suggest a modeling methodology that can integrate knowledge-based, resource-based, and environmental constraints, and which can proceed by a refinement strategy from incorporating major factors to incorporating subtler ones.

Interface designers often overlook aspects of an interface that facilitate operator errors, although in many instances the design problems are obvious once they are pointed out [Norman88]. Noticing these design problems becomes especially difficult in domains such as air traffic control where interfaces must mediate complex tasks carried out in diverse operating conditions. To help remedy this problem, we designed a human model specifically to help identify operating conditions in which controllers would be especially likely to make errors.

The details of our model, the Architecture for Procedure Execution (APEX), are discussed in [Freed96] and [Freed97]. APEX extends the CSS approach originally described by Remington et al. [Remington90, cf. Card83]. APEX combines two major components. The execution component provides the model with knowledge encoded as decision-making strategies needed to select action in a complex, dynamic, multitasking environment such as air traffic control. Selections are based largely on stored rules and procedures. The resource architecture represents a variety of human limitations such as the inability to look in more than one direction at a time and limited memory capacity. It constrains execution to operate within human limits. For example, if the execution component specifies a shift of gaze from one location to the other, the resource architecture will allow execution access to information about the new location but restrict information about the old.

The example below illustrates the ability of APEX to predict what are sometimes called "habit capture errors." The signature of a habit capture error is the execution of a habitual action in place of an intended but non-routine action. A common example of such an error might be the failure to stop at the store on the way home. The intent is formed before leaving work, but cannot be carried out until the car reaches the turn-off for the market. However, when that event occurs, instead of exiting the highway at the intended exit, the driver proceeds on the normal, habitual route. It would not be surprising to note that the frequency of such errors is high when drivers are very busy. Observation and anecdotal reports from air traffic controllers, however, indicate that habit capture errors occur at least as frequently during periods of low workload. A satisfactory account of habit capture errors then must explain how they arise in both high and low workload situations.

Example Air Traffic Control Scenario

At a Terminal Radar Control center, one controller will often be assigned to the task of guiding planes through a region of airspace called an "arrival sector" (For detailed discussions of air traffic controllers’ tasks, see [Stein93] and [Halverson95].) This task involves contacting planes at various sector entry points and getting them lined up at a safe distance from one another on landing approach to a particular airport. Some airports have two parallel runways. In such cases, the controller will form planes up into two lines (see Figure 1). Occasionally, a controller will be told that one of the two runways is closed and that all planes on approach to land must be directed to the remaining open runway. A controller's ability to direct planes exclusively to the open runway depends on remembering that the other runway is closed. How does the controller remember this important fact? Normally, the diversion of all inbound planes to the open runway produces an easily perceived reminder. In particular, the controller will detect only a single line of planes on approach to the airport, even though two lines (one to each runway) would normally be expected.

However, problems can arise in conditions of low workload. With few planes around, there is no visually distinct line of planes to either runway. Thus, the usual situation in which both runways are available is perceptually indistinguishable from the case of a single closed runway. The lack of perceptual support would then force the controller to rely on memory alone and thus increase the chance of error.

By helping to analyze such scenarios, APEX can direct an interface designer's attention to potential design-facilitated errors that might otherwise be overlooked. Moreover, the ability of APEX to make explicit how such errors might occur can help indicate the best way to refine an interface. For example, one of the difficulties in designing a radar display is balancing the need to present a large volume of information against the need to keep the display uncluttered. In this case, by showing how the error results from low traffic conditions, the model suggests a clever fix for the problem: Use an icon to explicitly represent runway closures, but only display the icon in low traffic conditions when it is most needed and produces the least clutter.

Some Principles and an Approach to Human-System Modeling

The basic requirements for APEX were (1) that it could model the performance of diverse tasks in complex task environments such as air traffic control, and (2) that its performance could vary in human-like ways depending on the design of its interface -- in particular, that it show approximately human tendency to err. Our model-building efforts were driven in part by careful analysis but also in part by trial-and-error. As patterns emerged regarding what would work and what would not, we inferred a set of general guidelines to help direct our efforts more effectively. In most cases, these guidelines made a great deal of sense in hindsight, but were not at all obvious at the outset. We present the approach that we eventually converged upon as a set of six principles, summarized below. We then discuss each in some detail.

1. Make the initial model too powerful rather than too weak.
2. Extend or refine the model only as required.
3. Model resource limitations and coping mechanisms together.
4. Use stipulation in a principled way.
5. Assume that behavior adapts rationally to the task environment.
6. Parameters that are of particular interest may be set to exaggerated values.

Discussion of Principles

Make the initial model too powerful rather than too weak

Human performance models are often evaluated by comparing their behavior to laboratory experimental data. For example, delays in responding to a stimulus in dual-task conditions exhibited by the CSS architecture [Remington90] closely approximate human delays in similar conditions. The high degree of fit between human and model performance is meant to provide evidence for the soundness and veridicality of these models. For these kinds of models, the need to characterize the details of human response-time distributions in simple, time-pressured tasks is of paramount importance.

For our purpose, the detailed accuracy of predicted response-time distributions must be weighed against the sometimes conflicting requirement that the model operate in a complex, multitasking domain. This conflict between accuracy and capability arises from limits on our scientific understanding of high-level cognitive tasks such as planning, task switching, and decision-making under uncertainty. To incorporate these capabilities into a model requires extensive speculation about how humans carry out such tasks, supplemented with knowledge-engineering in the domain of interest.

For models meant to be evaluated on the degree to which their performance fits empirical data, a reluctance to incorporate capable but speculative model elements is easily understood. Our goal of predicting performance in complex domains prescribes the opposite bias: If human operators exhibit some capability in carrying out a task, our model must also have that capability as a prerequisite to predicting performance at the task. One consequence of this bias is that our model may tend to be overly optimistic about human performance in some instances; the model performs effectively in situations where humans would fail. In our view, the increase in the model's ability to operate in interesting domains (where the need to predict design-facilitated error is greatest) outweighs the reduction in detailed accuracy.

Lacking an empirical basis for modeling certain cognitive activities, functional requirements and common sense have shaped the development of some aspects of the model. Moreover, we have borrowed and adapted decision-making mechanisms developed by artificial intelligence researchers to serve as our model's execution component. The executive, derived from a robot control language called RAPs [Firby89], extends the action-selection capabilities provided by CSS and GOMS in ways that have a great deal of significance for modeling performance in air traffic control. These extension include:

- ?? effective coordination of perceptual, cognitive, and motor resources
- ?? diverse mechanisms for handling task interruption, switching, and resumption
- ?? planning mechanisms to cope with inherently dynamic and uncertain aspects of the task environment
- ?? ability to monitor for and recover from plan failure

We discuss this component of the model at greater length in [Freed96], though see [Firby89] for a detailed account of its use and capabilities.

Extend or refine the model only as required

Early versions of our current model were developed with the idea that any reliable psychological finding that could be incorporated into the model constituted a useful addition. Our initial goal was thus to bring together as much psychology, neuroscience, anthropometry, and so on as possible. Over time, we found many occasions in which elements of the model added insufficient value to compensate for difficulties they created.

For example, early versions of the model incorporated the finding that human vision takes slightly longer to process certain perceptual features than others; color, for instance, takes a few milliseconds longer to process than orientation or primitive shape. Of the kinds of predictions our model could reliably make, none depended on this aspect of the model. In fact, we found it difficult even to imagine situations in which useful predictions would arise from this element. Moreover, its inclusion was quite costly since it forced simulation mechanisms to consider very brief time intervals (one millisecond), thus slowing simulations substantially.

Adding unnecessary detail to the model makes it model slower in simulation, increases the amount of effort needed to make future improvements, and makes it harder to debug, explain, and evaluate. Therefore, it is important to make sure that extensions to the model make it more useful. In our case, that means it should help us in highlighting opportunities for operationally significant human error.

Another instructive example concerned our efforts to model time delays in acquiring information for a decision task. For example, a controller deciding which runway to direct a plane towards must acquire information on such factors as the relative number of planes lined up for each alternative runway, the weight of the plane (Heavy planes should preferably be sent to the longer runway.), and whether each runway is operational. To acquire information about any of these factors from memory, a controller would have to employ his/her memory retrieval resource which can only be used for one retrieval task at a time [Carrier95].

Since use of the retrieval resource blocks its availability to other decision-making tasks (and also delays the current decision task), the amount of time required to perform a retrieval can be an important determiner of overall performance. Incorporating the determinants of retrieval time into the model would thus seem to have great value in predicting performance. However, two other factors suggest the need for care in deciding what aspects of memory retrieval should be modeled. First, a survey of the literature on memory reveals numerous factors affecting retrieval time. Incorporating each of these factors into the model would likely involve a lifetime of effort.

Second, controllers typically have alternative ways to evaluate the factors that bear on their decisions, each varying in required time and other properties. For example, to acquire information about the weight class of a plane, a controller can (a) read the weight value off the plane’s data block on the radar display, (b) retrieve that plane’s weight from memory, or (c) assume that the plane has the same weight class as most other planes. The time required to carry out these methods can differ by orders of magnitude. In our model, relying on a default assumption requires no time or resources; memory retrieval requires approximate .5 seconds; visual search and reading require a highly variable amount of time ranging from 0.5 seconds to 10 seconds or more. Based on the magnitude of these differences, we have assumed that model refinements that increase our ability to predict which information acquisition method will be used are generally more valuable than refinements that account for variance in memory retrieval time.

There are two corollaries to the principle of letting modeling goals drive refinement efforts. First, as illustrated by the example of modeling differential propagation rates of low-level visual features, one should prefer to maximize the temporal coarseness of the model with respect to the desired classes of predictions. Model elements that rely on temporally fine-grained activities should be included only if their inclusion accounts for significant differences in overall task performance. Second, as illustrated by the memory modeling example, prefer to model the largest sources of performance variability in a given activity before modeling smaller sources.

Model resource limitations and coping mechanisms together

In our view, much of people's tendency to err can be explained as a consequence of limitations on perceptual, cognitive, and motor resources. We now believe, however, that the most obvious ways of linking errors to resource limitations are misleading. In particular, each limitation can be associated with a set of behaviors used to cope with that limit. These coping behaviors rely on assumed regularities in the world and on other assumptions that can sometimes prove false. The imperfect reliability of a coping method's underlying assumptions renders people susceptible to error. This is something of a reconceptualization of the problem, as it moves the problem locus from peripheral resources which are somehow “overrun” by task demands, to the decision making and plan execution component of the model.

For example, people cope with a restricted field of view by periodically scanning their environment. Mechanisms for guiding the scan must guess where the most interesting place to look lies at any given time. By making some assumptions, for example, that certain conditions will persist for a while after they are observed, scanning mechanisms can perform well much of the time. But even reliable assumptions are sometimes wrong. People will look in the wrong place, fail to observe something important, and make an error as a result. Of course, people have no choice about whether to scan or not; if a person were somehow prevented from scanning, many tasks would be impossible. By forcing people to guess where to look, a limited field of view enables error.

Human resource limits are much easier to identify and represent in a model than are the subtle and varied strategies people use to cope with those limits. For example, people have limited ability to ensure that the things they have to remember “come to mind” at the right time. Modeling this requires the separation of the processes that determine the result of a retrieval attempt from those that initiate a memory retrieval attempt. Retrieval initiation happens only when triggered by certain conditions external to the memory model itself.

People cope with memory limitations by maintenance rehearsal, writing notes to themselves, setting alarms, and other methods. Unless the model includes mechanisms needed to carry out these strategies, it will tend to under-predict human performance -- that is, it will predict failures of memory where people would not actually fail. As discussed above, our purposes require exaggerating rather than understating expert performance when an accurate model is not possible.

Use stipulation in a principled way

While it is challenging and scientifically worthwhile to show how intelligent behavior can emerge from the harmonious interplay of myriad low-level components, practical considerations require us to model these low-level component processes abstractly. In some cases, the need for abstract process models arises from the practical considerations already discussed -- that is, to avoid complicating the model with elements that add little to its power to make useful predictions. In other cases, scientific ignorance about how processes are carried out forces us to stipulate that a process occurs without specifying any mechanism.

For example, in designing model elements representing human vision, we had to contend with the fact that no complete and detailed model of human visual processing currently exists; in fact, no existing model of visual processing, including robot vision systems designed without the requirement that they conform to human methods or limitations, can achieve anything close to human performance at tasks like dynamic pattern recognition and visually guided navigation. Thus, we could not have represented the mechanism of normal visual function, even if doing so would have been worthwhile in terms of previously described goals.

Instead, our model requires that the simulated controller operate in a perceptually simplified world in which a detailed representation of the visual scene, for example, as an array of intensity- and chroma-valued pixels, is abandoned in favor of qualitative propositions representing properties of visible objects. For instance, to represent planes observable on a radar display, the world model generates propositions such as

```
(shape visual-obj-27 airplane-icon .2)
(color visual-obj-27 green .5)
(location visual-obj-27 135 68 .1)
```

which together represent a green plane icon located at a given position relative to a reference point.

To simulate nominal visual performance, the vision model simply passes propositions from the world to cognitive model elements. Thus, decision-making elements would, in some cases, simply be informed that there is an object at location `(135,68)` without vision having to derive this information from any more fundamental representation.

Given our goal of accounting for the effect of interface attributes on controllers' performance, the need to eliminate any explicit representation of visual processing poses an important problem: How can we account for the effect of interface attributes such as color, icon shape, and the spatial arrangement of visual objects except by allowing them to affect processing? To illustrate our approach, consider how the model handles direction of gaze, one of the most important determinants of what visual information is accessible at a given moment.

The first step was to construct a basic model of visual processing that would successfully observe every detail of the visual environment at all times. As described, this simply required a mechanism that would pass propositions describing the visual scene from the world model to cognitive mechanisms. Real human visual performance is, of course, limited to observing objects in one's field of view. Moreover, the discriminability of object features declines with an object's angular distance from fixation (the center of gaze). To model this, we require that propositions generated by the world model include information on the discriminability of the visual feature each represents. For example, the proposition

```
(shape visual-obj-27 airplane-icon .2)
```

means that `visual-object-27` can be recognized as an `airplane-icon` as long as its distance in angular degrees from the agent's point of fixation lies within a certain range. In this case, the given proposition becomes accessible to cognition if features subtending `.2` degrees of visual angle can be discriminated at the object's current distance from fixation.

This is an unusual way of looking at the process of acquiring information, though similar to the approach used in MIDAS [Corker93]. Rather than modeling the process of constructing information from perceptual building-blocks, all potentially relevant information items are considered potentially available; we simply stipulate that the constructive processes operate successfully. The task of the model then is to determine which potentially available items are actually available in a given situation.

A generalization of this approach is also used in non-perceptual components of the model. In general, nominal performance is stipulated, and factors that produce deviations form nominal performance are modeled separately and allowed to modulate nominal performance.

Assume that behavior adapts rationally to the environment

For most performance variables of interest, the amount of practice constitutes the single largest source of variability. People become adapted to their task environment over time, gradually becoming more effective in a number of ways [Anderson90, Ericsson91]. For the purpose of modeling highly skilled agents such as air traffic controllers, this process has several important consequences.

First, people will, over time, come to learn about and rely on stable attributes of the task environment. For instance, the air traffic control scenario discussed earlier, the controller relied on the (false) default assumption that both runways were available. Constructing a model to predict such an error thus requires determining that certain conditions are much more common than others and are likely to be treated as default assumptions by experienced operators. Similarly, the controller in our example relied on a perceptual cue, a linear arrangement of plane icons on the radar display, to signal that a non-default condition might hold and that a memory retrieval action was warranted. Thus our model requires determining what kinds of perceptual cues are likely to be available in the environment and to be exploited by experienced operators to support cognition.

A second consequence of adaptation that should be considered in the construction of models such as APEX is the fact that, over time, people will learn which policies and methods work and which tend to fail. This significantly complicates analyses of the effect of innate human resource limitations on task performance. For example, experienced grocery shoppers will come to learn that relying on memory to retain the list of desired goods does not tend to work very well. Experienced shoppers will almost inevitably come to rely on some strategy that circumvents the limitations on their memory [Salthouse91]. For example, some will rely on a written list; others might learn to scan the shelves for needed items, thus replacing a difficult memory task (recall) with an easier one (recognition).

To account for the effect of limitation-circumventing strategies, our model includes a variety of mechanisms for representing the proceduralized behaviors that incorporate these strategies. For instance, procedures representable in our model can integrate physical and cognitive actions to carry out visual search tasks that result in the initiation of a memory retrieval followed by a decision-making task that depends on the output of the memory retrieval. However, the ability to represent such procedures is not enough to enable us to predict the performance of experienced practitioners of a task. It is also necessary to know what strategies they will tend to employ, and thus what procedures should be represented. We discuss this issue in [Freed96].

The assumption that experienced practitioners will have adapted to their task environment provides a basis for setting otherwise free parameters in the model. For example, our account of prospective memory, a key element of the model for predicting habit capture errors, assumes that the likelihood that a person will attempt to verify a default assumption by retrieving information from memory declines over time. For example, the air traffic controller in the example scenario became less likely to retrieve knowledge about the runway closure from memory as time elapsed since the last time s/he was reminded of the closure by a visible anomaly on the radar display.

Constructing the model required making some assumption about the rate at which retrieval likelihood would decline. Note that this value could, in principle, be obtained in a controlled experiment. However, performing such an experiment would undermine the whole purpose of the modeling effort which is to provide performance estimates in the absence of empirical testing. Our approach was to assume that the retrieval-attempt likelihood function depended only on considerations of utility, and not on any innate limitations.

In particular, we considered three factors. First, air traffic controllers must generally learn to minimize the use of limited cognitive resources in decision-making to cope with potentially very high workload. Thus, optimal decision-making performance must avoid memory retrieval whenever the result is likely to confirm a default assumption.

Second, regularities in the duration of a given non-default condition indicate that, after a certain interval, decision-mechanisms can once again reliably assume the default.

Third, regularities in the rate at which perceptual indicators of the non-default condition are observed can provide an accurate determination of when the default condition has resumed -- that is, if an indicator is usually observed within a given interval, the absence of that interval for the interval can be treated as evidence for the default.

We estimate memory-retrieval likelihood

```
mrl = min[D(p),I(p)]
```

where `D(p)` is the maximum duration of the non-default interval with likelihood (confidence) `p`, and `I` is the maximum interval between successive observations of a non-default indicator with likelihood `p`.

We note that functional estimates of this sort are famous for producing bad theories in certain areas of science such as evolutionary biology. But in the absence of extensive empirical research, the assumption that parameters will have been set by some optimizing adaptive process [Anderson90] will often be a good approximation and will usually constitute the most conservative available guess.

Parameters that are of particular interest may be isolated or set to exaggerated values.

Our purpose involves highlighting vulnerability to human error in complex, dynamic domains. Like many other domains where predicting design-facilitated error would be useful, operating in air traffic control requires a powerful (highly capable) model of how actions are selected. Furthermore, the air traffic control system is operated by highly skilled individuals, and the system itself is designed to prevent or manage errors with extremely high success rates.

For our purposes, it is not particularly useful to simulate the actual (almost negligible) error rates of the existing air traffic control system. Therefore, once we have built a capable model and selectively introduced constraints and coping mechanisms, the user should be able to choose parameter values that exaggerate the simulated operator's vulnerability to error. For example, the model may be parameterized with unrealistically pessimistic assumptions about working memory capacity, in order to exaggerate the dependence upon perceptual sources of information. Lewis and Polk [Lewis94] used this technique to model an aviation scenario in Soar in such a way as to highlight the need for perceptual support: they used a Soar model with a zero-capacity working memory.

The need for this bias stems from the fact that a designer is usually interested in counteracting even low probability errors, especially when the consequences of error are high or where the task will be repeated often. If low probability errors only showed up in simulation with low probability, the model would often fail to draw attention to important design flaws.

Conclusion

In modeling something as complex and difficult to specify as a human operator, decisions about how to direct model development effort have enormous impact in determining the utility of the resulting model. We have come to believe that the intended application of the model should play a dominant role in making such decisions. Our purpose in this case was to highlight the risk of certain kinds of human error. The principles we have developed -- although each one has arguably been stated by someone at some time -- are often counter-intuitive; for example, put as little into the model as possible, not as much as possible; perhaps set key parameters to intentionally unrealistic values, rather than investing time and effort to estimate "correct" values. Taken together, these principles suggest a methodology for the development of human performance models for complex human-machine systems: what to do first, what to do later, and what not to do at all.

References

[AMODEUS97] AMODEUS Project. (1997). The AMODEUS Project World Wide Web (WWW) site. http://www.mrc-apu.cam.ac.uk/amodeus/

[Anderson90] Anderson, J.R. (1990). *The adaptive character of thought*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[Card83] Card, S.K., Moran, T.P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[Carrier95] Carrier, L.M., & Pashler, H. (1995). Attentional limitations in memory retrieval. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 21, 1339-1348.

[Corker93] Corker, K.M., & Smith, B.R. (1993). An architecture and model for cognitive engineering simulation analysis. In *Proceedings of the AIAA Computing in Aerospace 9 Conference.* San Diego, CA: AIAA.

[Degani96] Degani, A. (1996). *Modeling human-machine systems: On modes, error, and patterns of interaction*. Ph.D. thesis. Atlanta, GA: Georgia Institute of Technology.

[Degani97] Degani, A., & Shafto, M.G. (1997). *Process algebra and human-machine modeling*. Cognitive Technology Conference. Terre Haute, IN: Indiana State University.

[Ericsson91] Ericsson, K.A., & Smith, J.A. (Eds.). (1991). *Toward a general theory of expertise.* Cambridge, UK: Cambridge University Press.

[Firby89] Firby, R.J. (1989). *Adaptive execution in complex dynamic worlds*. Ph.D. thesis, Yale University, New Haven, CT.

[Freed96] Freed, M.A. (1996). Using the RAP system to predict human error. In *Proceedings of the 1996 AAAI Symposium on Plan Execution: Problems and Issues*. AAAI Press.

[Freed97] Freed, M.A. & Remington, R.W. (1997, forthcoming). Managing decision resources in plan execution. In *Proceedings of the Fifteenth Joint Conference on Artificial Intelligence.* Nagoya, Japan.

[Gould88] Gould, J.D. (1988). How to design usable systems. In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. New York: North-Holland.

[Gray93] Gray, W.D., John, B.E., & Atwood, M.E. (1993). Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world task performance. *Human Computer Interaction*, 8, 237-309.

[Halverson95] Halverson, C.A. (1995). *Inside the cognitive workplace: New technology and air traffic control*. Unpublished doctoral dissertation. La Jolla, CA: University of California, San Diego.

[John94] John, B.E., & Kieras, D.E. (1994). *The GOMS family of analysis techniques: Tools for design and evaluation*. Carnegie Mellon University, School of Computer Science, TR CMU-CS-94-181.

[Landauer95] Landauer, T.K. (1995). Let’s get real: A position paper on the role of cognitive psychology in the design of humanly useful and usable systems. In R.M. Baecker, J. Grudin, W.A.S. Buxton, & S. Greenberg (Eds.) *Readings in Human-computer Interaction: Toward the Year 2000*. Los Altos, CA: Morgan Kaufman.

[Lewis94] Lewis, R.L., & Polk, T.A. (1994). *Preparing to Soar: Modeling pilot cognition in the Taxi-MIDAS Scenario*. Cognitive Modeling Workshop. Moffett Field, CA: NASA-Ames Research Center.

[May94] May, J., Blandford, A., Barnard, P., & Young, R. (1994). *Interim report on the application of user modelling techniques to the shared exemplars*. AMODEUS Project Document D6. Cambridge, UK: MRC Applied Psychology Unit.

[Norman88] Norman, D.A. (1988). *The psychology of everyday things*. New York: Basic Books.

[Reason90] Reason, J.T. (1990). *Human error*. Cambridge, UK: Cambridge University Press.

[Remington90] Remington, R.W., Johnston, J.C., Bunzo, M.S., & Benjamin, K.A. (1990). The cognitive simulation system: an interactive graphic tool for modeling human cognitive reasoning. In *Objected-oriented Simulation*. San Diego, CA: Society for Computer Simulation.

[Remington90a] Remington, R.W., & Shafto, M.G. (1990). *Building human interfaces to fault diagnostic expert systems I: Designing the human interface to support cooperative fault diagnosis*. Seattle, WA: CHI'90 Workshop on Computer-Human Interaction in Aerospace Systems.

[Salthouse91] Salthouse, T.A. (1991). Expertise as the circumvention of human processing limitations. In Ericsson, K.A., & J.A. Smith (Eds.), *Toward a general theory of expertise* . Cambridge, UK: Cambridge University Press.

[Seifert94] Seifert, C.M., & Shafto, M.G. (1994). Computational models of cognition. In J. Hendler (Ed.), *Handbook of Cognitive Neuropsychology*, vol. 9. Amsterdam: Elsevier.

[Shafto90] Shafto, M.G., & Remington, R.W. (1990). *Building human interfaces to fault diagnostic expert systems II: Interface development for a complex, real-time system*. Seattle, WA: CHI'90 Workshop on Computer-Human Interaction in Aerospace Systems.

[Shafto94] Shafto, M.G., Remington, R.W., & Trimble, J.P. (1994). *A refinement framework for cognitive engineering in the real world*. Paper prepared for the symposium "Cognitive Science Meets Cognitive Engineering," Cognitive Science Society Annual Conference. Atlanta, GA: Georgia Institute of Technology.

[Stein93] Stein, E. S., & Garland, D. (1993). *Air traffic controller working memory: Considerations in air traffic control tactical operations*. FAA technical report DOT/FAA/CT-TN93/37.